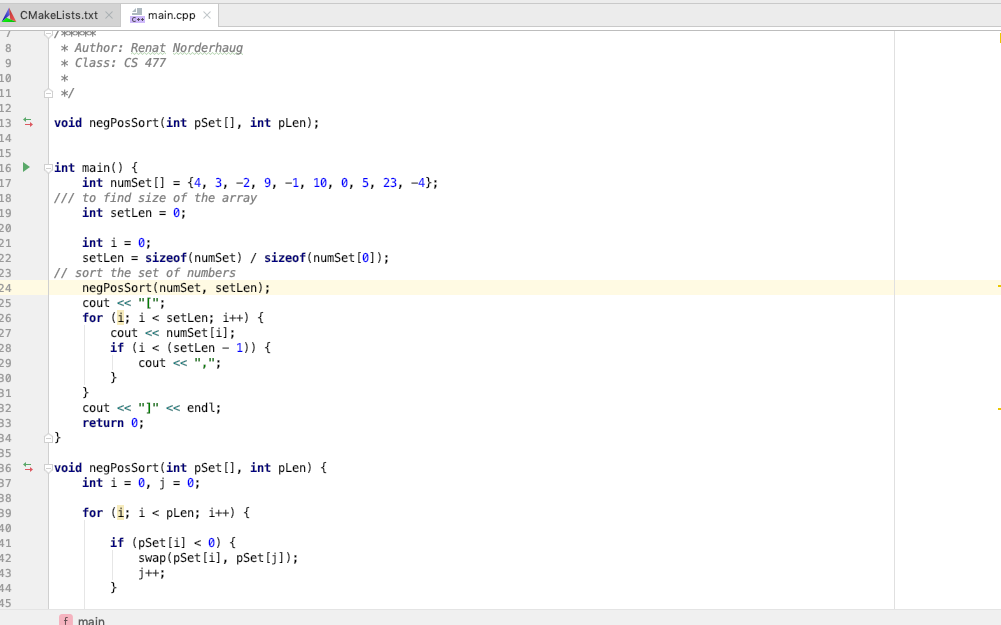
Renat Norderhaug

CS 477

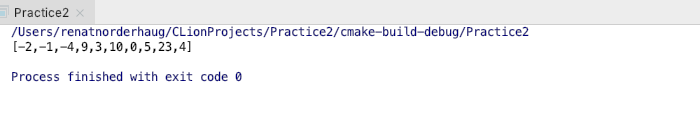
Homework 4

1. Implement in C/C++ an algorithm to rearrange elements of a given array of n real numbers so that all its negative elements precede all its positive elements. Your algorithm should be both time- and space-efficient. Show how your algorithm works on the following input: A = [4 3 -2 0 2 9 -1 10 0 5 23 -4]

**code:**

****

output:



2. Answer the following question: is Quicksort a stable sorting algorithm? If yes, give a justification. If not, provide a counterexample.

**Merge Sort is an example of a stable sorting algorithm. Quicksort is not a stable sorting algorithm because an algorithm is considered stable if it can maintain the order of elements when considering equality of keys. More specifically quicksort is not a stable sorting algorithm because elements in quicksort are swapped using pivot positions which consequently ignores the original positions of elements. Insertion sort is stable because the order of equal elements do not change when sorted.**

6. Exercise 9.3-3

Assuming all elements are distinct, we can change the quicksort to run in O(nlogn) by changing the partition prodecure. We can make the partition procedure use the select algorithm to use the median of the array as a pivot element so we can ensure the input will split in half which gaurantees best-case partition. We set up the following recurrence to find the find the run time:

T(N) = 2T(N/2) + O(n) = O(nlogn)